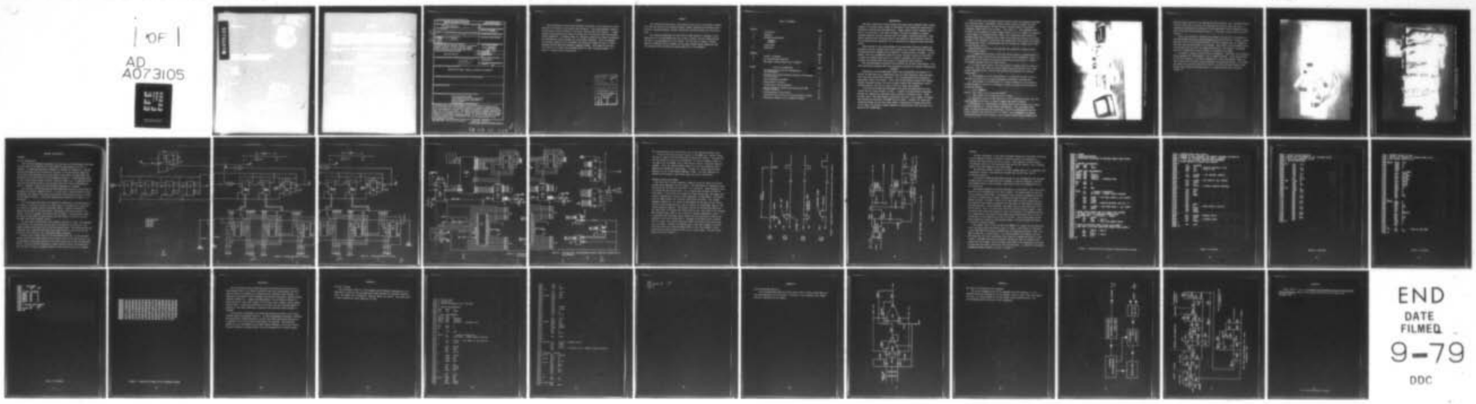
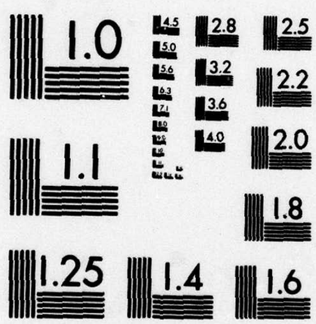
AD-A073 105   AEROSPACE MEDICAL RESEARCH LAB WRIGHT-PATTERSON AFB OH   F/G 9/2
DESIGN OF A MICROPROCESSOR BASED CARDIOTACHOMETER. (U)
APR 79 D RATINO, G POTOR, A MARKO, C SHARPER
UNCLASSIFIED   AMRL-TR-79-21   NL

| OF |
AD
A073105



END
DATE
FILMED
9-79
DDC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER 14 AMRL-TR-79-21

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER 9

4. TITLE (and Subtitle) 6 DESIGN OF A MICROPROCESSOR BASED CARDIOTACHOMETER,

5. TYPE OF REPORT & PERIOD COVERED Technical Report,

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s) 10 D. Ratino, C. Sharper, G. Potor, A. Marko

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Aerospace Medical Research Laboratory, Aerospace Medical Division, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio 45433

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
62202F, 6893-07-01 16 17

11. CONTROLLING OFFICE NAME AND ADDRESS
(same as block 9)

12. REPORT DATE 11 April 1979

13. NUMBER OF PAGES 36

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)
12 39p.

15. SECURITY CLASS. (of this report) UNCLASSIFIED

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Electrocardiogram (EKG)
Microprocessor Based Cardiotachometer
Beat-by-BeatHeartrate Measurement
Digital Readout
Microprocessor Instrumentation

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A miroprocessor based cardiotachometer has been designed and fabricated in-house using 8080 Intel microcomputer hardware and assembly language software. The cardiotachometer accurately calculates and digitally displays beat-by-beat hear rate in beats per minute. The design incorporates standard semiconductor memory and I/O integrated circuits. Operational understanding does not require extensive microprocessor design background. In fact, the cardiotach's design provides excellent basic information in the use of microprocessor in the instrumentation field.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

009 850

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

79 08 27 050 mt

SUMMARY

The microprocessor based cardiotachometer described in this report was tested and found to function very reliably and accurately. The cardiotachometer requires no calibration and yet produces an errorless digital heart rate readout. A simple modular design approach was achieved through the utilization of 8080 Microcomputer system compenents. A minimum number of integrated circuits were used to accomplish the interfacing, timing, memory, and input/output functions. Both the hardware and software aspects of the instrument are explained through the use of circuit diagrams and the annotated assembly language program. The reason for writing this report is to convey the knowledge gained to other interested people who desire to design with microprocessors.

PREFACE

The research and development described here was carried out between September 1977 and June 1978 in the Simulation Support Branch, Manned-Systems Effectiveness Division, Aerospace Medical Research Laboratory, Wright-Patterson Air Force Base, Ohio.

The authors acknowledge the assistance of TSgt Gregory D. Bathgate in the fabrication of the instrumentation, Mrs. Lynda A. Powell, and Mrs. Margy Horvath for help in the preparation of this report. We also wish to thank Mr. Donald McCollor, Raytheon Service Company, for his power supply design suggestions and the inclusion of his electro-cardiogram amplifier in the appendix.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

INTRODUCTION

The rapid evolution of microprocessor and large scale integrated (LSI) circuit elements is causing a major change of philosophy in the design and construction of electronic apparatus and instrumentation. Highly complex circuit modules, available at low cost and in very small packages, has made it possible to package complete computers in small portable units for single, dedicated functions. This report resulted from the efforts made by the authors to gain experience with the use of microprocessor components in the construction of special purpose instrumentation.

Since details of the inner-workings of microprocessor chips are complex and frequently obscure, designs using these elements must proceed "cook-book" fashion, using manufacturers suggestions and samples of working products as models. At the outset, there was considerable skepticism about the earlv success of this effort. A project was needed which would be as simple as possible, yet utilize computer concepts and components and also result in a needed and useful product.

The cardiotachometer described here fulfilled all of these requirements. Circuits incorporate designs taken from information supplied by component manufacturers.

APPROACH

Obtaining an electrocardiograph signal and electronically processing it to extract the QRS event as a standardized pulse of fixed amplitude, shape and duration, uses well established techniques. The following method was proposed for developing a heart rate display. Start with the QRS signal, measure the interval between successive pulses, from this interval compute the instantaneous rate in beats-per-minute, and use the result to drive a three digit display.

A facility for design and development work had to be established for this project and for future projects as well. A number of microprocessor development systems are commerically available using a variety of processor chips. These systems cover a very wide range of flexibility and cost. For a number of reasons, some necessarily arbitrary, the 8Ø8Ø chip was chosen. Probably the single most compelling reason was the very large number of manufacturers making parts based on this chip, all within a single framework of compatability which has come to be known as the "S-1ØØ Buss".

Thus, we built our development system around an IMSAI microcomputer mainframe (with operator panel), 16K memory, Tarbell Format Cassette Storage, Processor Tech ALS-8 Operating System, CRT and Teletype terminals and an "Intelligent Breadboard" (Model BBC-5) also by IMSAI. (see Figure 1). The breadboard forms the heart of the system for circuit design purposes, as it is capable of accepting a large number of integrated circuit chips of all shapes and sizes along with necessary capacitors, resistors, etc.; while providing direct access to the inner workings of the computer. In this way, the computer not only controls the signals for testing new circuit configurations, but also becomes an integral part of the circuits as they evolve.

The architecture which was developed for the cardiotach consists of the following main elements:

1. An 8Ø8Ø central processor chip (CPU) which executes a program stored in a 27Ø8 EPROM, using a 681Ø RAM chip for 128 bytes of 8-bit read/write memory. An 8228 is used as the system controller and bi-directional bus driver. A 8224 is the systems' crystal controlled master clock.

2. A 16-bit binary counter (two 8-bit chips in tandem) driven to count at a 1 KHz rate. The 1 KHz signal, derived by decade countdown from the CPU'S 2 MHz crystal controlled clock, serves as the precise time reference for counting milliseconds.

3. The program runs in a loop waiting for a pulse from the QRS circuits. This pulse is received via a 8255A programmable peripheral interface chip.

4. Upon receipt of the QRS pulse, the counter contents are transferred to the CPU whereupon a division is performed to obtain the heart rate according to the trivial formula:

H R = 6ØØØØ/MS

Where - - HR = heart rate in beats per minute

MS = number of millisecond counts per beat.

5. The binary results of this division is then converted to three 4-bit BCD digits and fed via an 8255A to three 7-segment LED numeric displays.

The procedure for implementing this design started with setting up the components for the counters, display LED and interfacing on the intelligent breadboard, to allow development of the software. The first configuration used two 8255's - one for input and the other for output. After debugging the program sufficiently to get correct results, various modifications in both software and

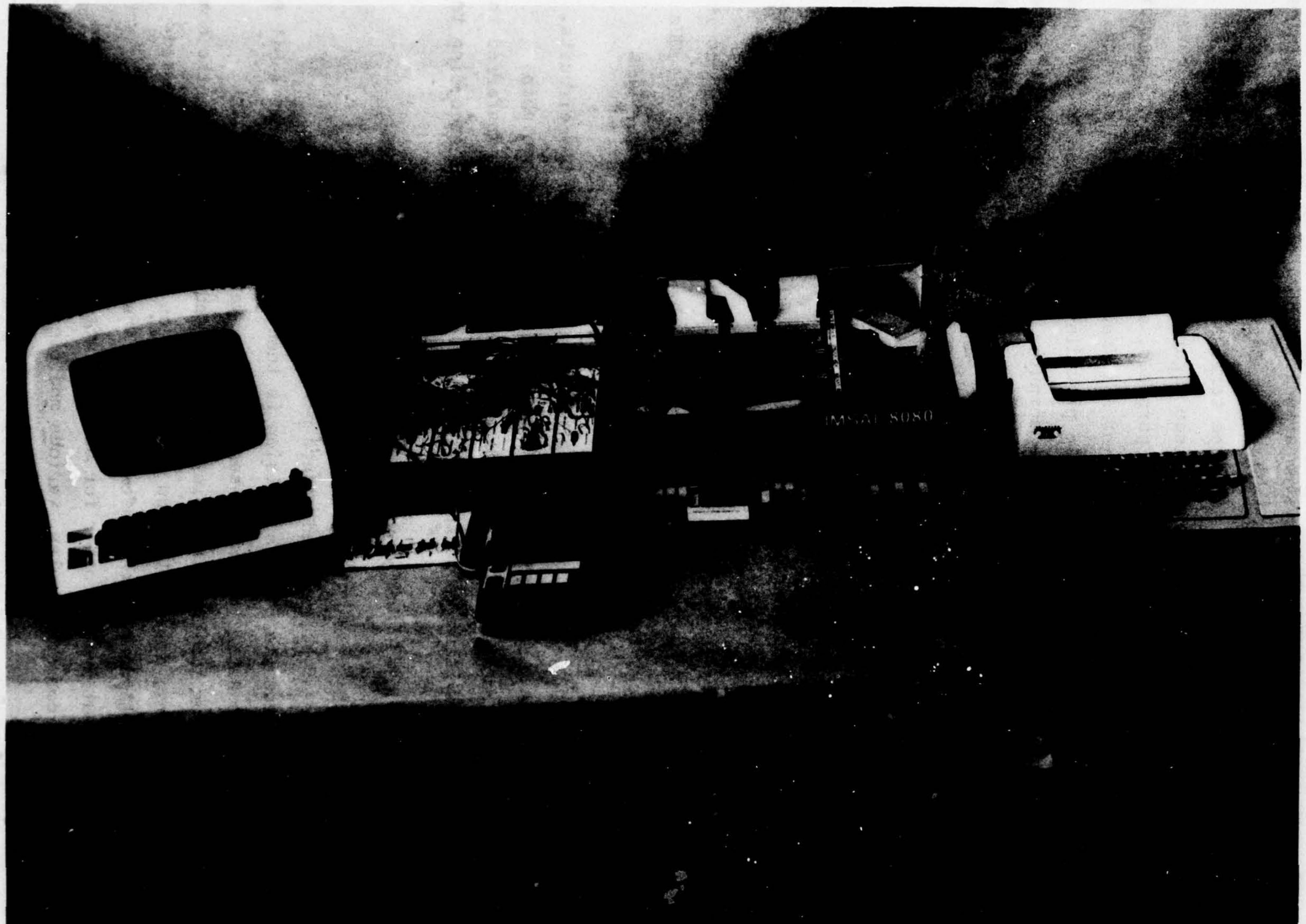


Figure 1. The experimental setup for designing the cardiotachometer.

hardware details were tried to simplify the entire system. This included eliminating one of the 8255's and using the remaining one for both input and output. A stand-alone experimental prototype of the microprocessor system hardware was then built using wire-wrap techniques (Figure 2). This was a crucial step for several reasons.

First, the interactive terminal and software aids would not be available in the wire-wrapped prototype, making debugging much more difficult. Second, a great majority of the circuit features needed in the full computer were not needed for this unit and were eliminated in this design. The results of these changes would not be known until the unit was completed and tried. And finally, the program would be on EROM and not easily subject to examination or change.

As expected, results of first tests were negative. Very short test precedures were programmed into 27Ø8 EPROMS to aid in signal tracking. (The listing of one of the test programs is included in Appendix A). After considerable labor over what turned out to be minor circuit omissions, the prototype worked. It was used as a model for the construction of the final unit which included etched circuit boards, power supply and display panel in a complete stand-alone enclosure (Figure 3).

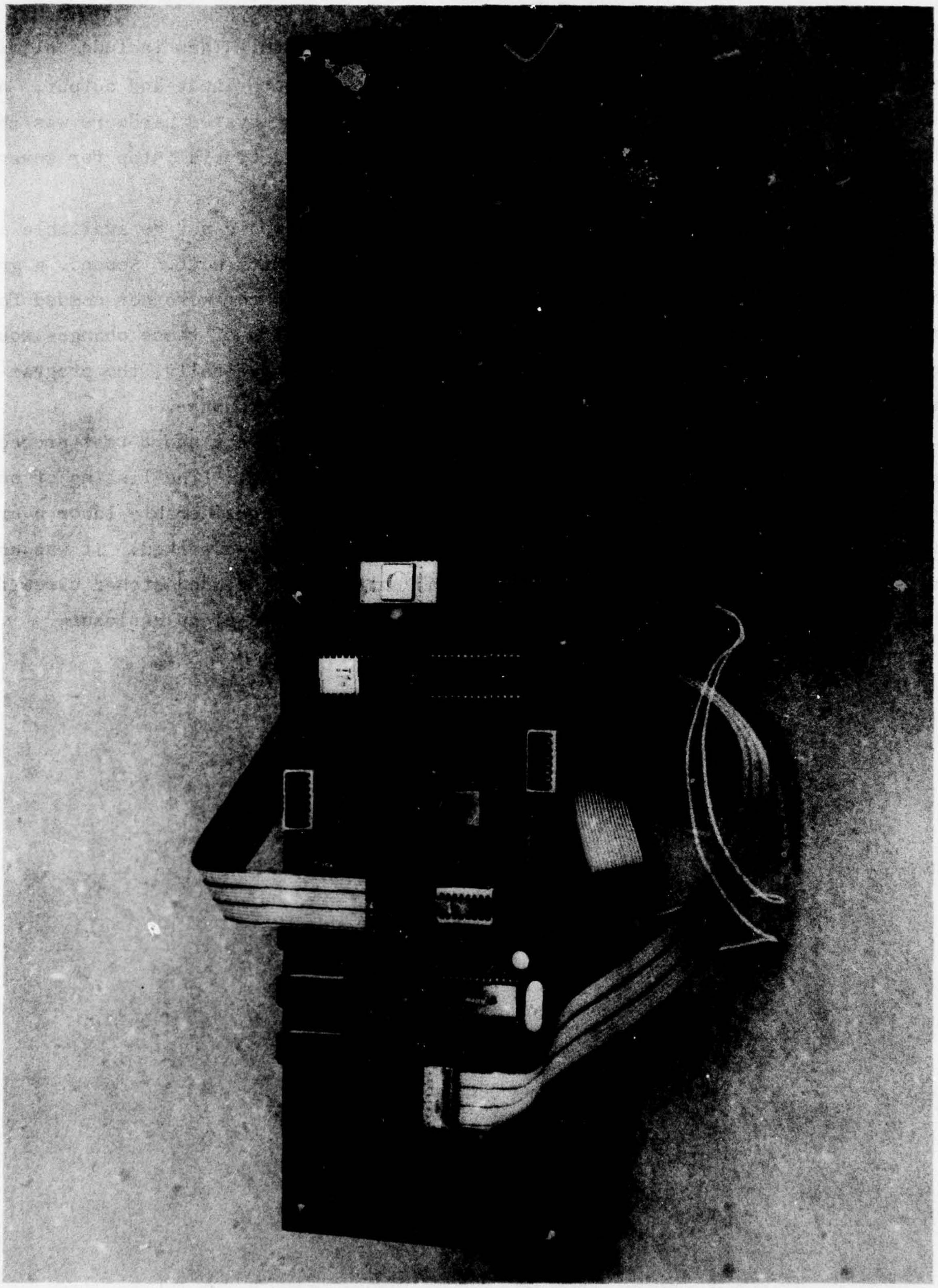


Figure 2. Microprocessor system hardware prototype wire-wrapped on vector board.

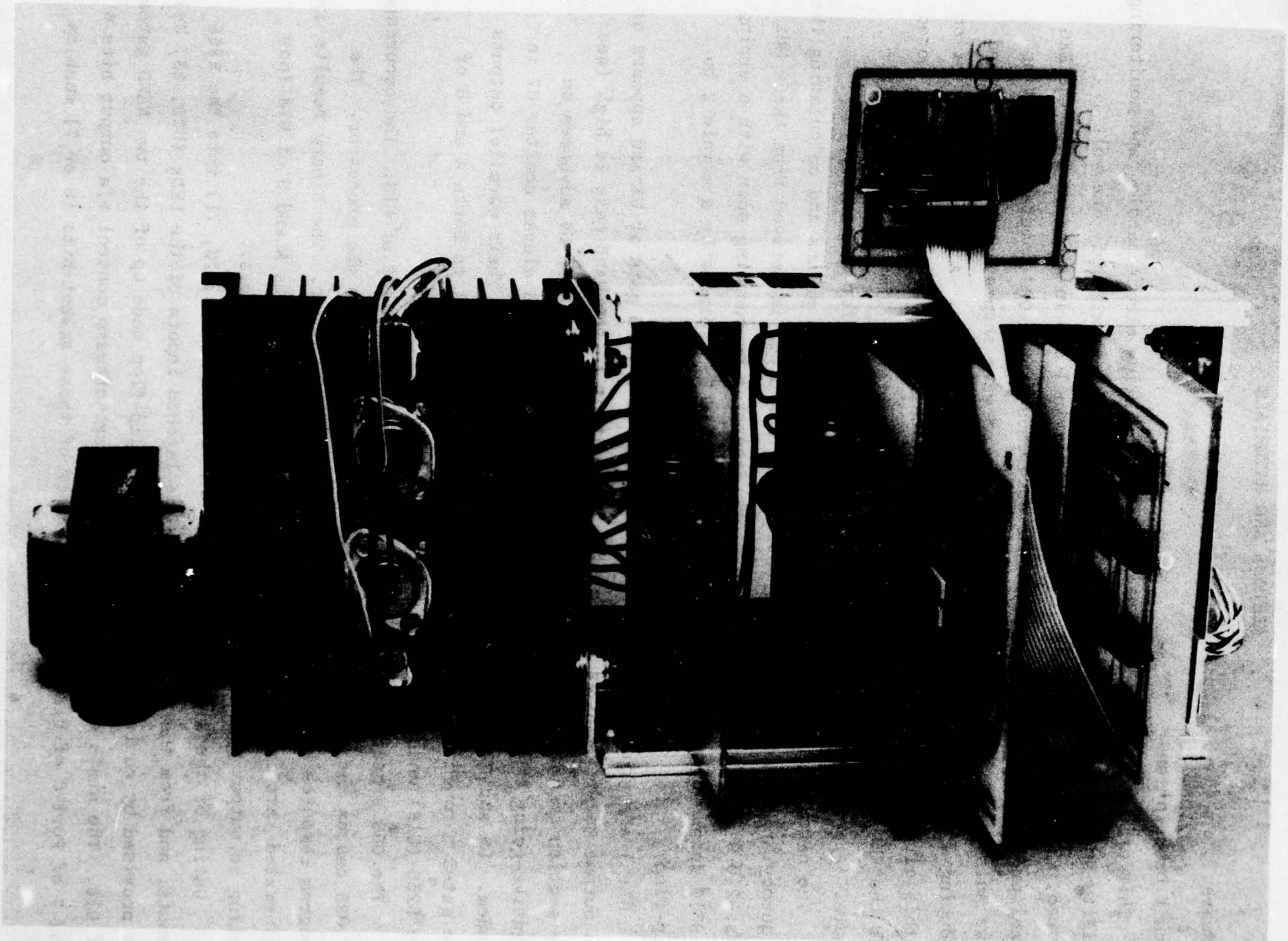


Figure 3. Cardiotachometer card rack showing power supply card, EKG signal processing card, 8080A CPU card, P8255A I/O card, and counter/buffer card.

METHODS AND MATERIALS

Hardware

Circuit Description

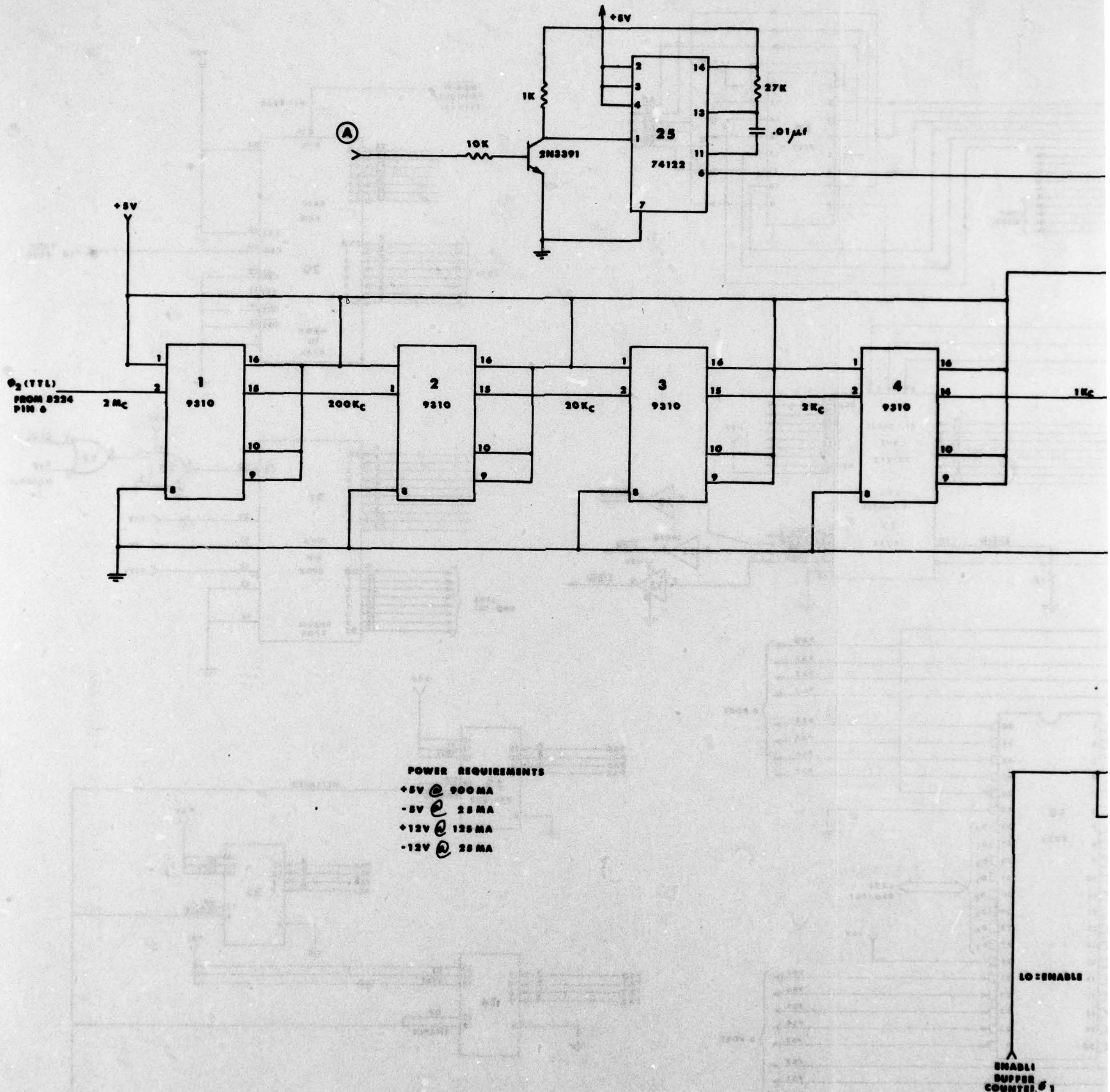
This unit was designed to operate in an existing electrocardiograph monitoring facility and the availability of an EKG signal of approximately one volt is assumed. (A simple EKG amplifier schematic is included in Appendix B.) The EKG signal processing section extracts the occurence of the QRS complex, rejecting noise pulses and operating over a wide range of wave shapes ana amplitudes, and producing a $\pm$2.5 volt pulse of 125 milliseconds duration. A detailed description of this circuit may be found in another technical report (1). It should be noted that the EKG processing section has an upper heart rate limit of 300 beats per minute. Appendix C shows block and schematic diagrams for reference.

The circuitry for determining the time between QRS pulses and calculating the frequency are shown in Figures 4 and 5. A 1 KHz clock is derived from the 2 MHz phase-2($\emptyset_2$) TTL output of the 8224 clock generator by counting down with a string of 931Ø decade counters. (The fourth 931Ø could be replaced by a simple 2 to 1 flip-flop).

The 1 KHz square wave from U4 is applied to a NAND gate at U5 and appears at its output to drive U6 as long as the other input to the NAND gate is high (see the section of Sequence Control and Timing, below). Counter U6 advances on negative-going edge of the clock pulse. The output of U6 clocks counter U7 in tandem. U6 and U7 together thus form a 16-bit counter. Their parallel outputs are gated through 8T97 tri-state buffers U9, U10, and U11 to Ports A and B of the 8255 (U18) under software control.

The QRS signal is monitored by the software via Port C of U18. The occurence of QRS causes the count since the last QRS to be input to the processor. The program computes beats per minute from the interval count. The binary results are reformated into individual BCD digits and output via Ports A and B of U18 (now acting as output ports) to the latched LED numeric displays.

Gating of the signals from the counters buffers (U9, 1Ø, 11) into the 8255 inputs, and from the 8255 outputs to the latched inputs of the LEDs (U22, 23, 24), is provided by control levels from the RS flip-flop made up of the two NAND gates of U15. The state of this flip-flop is under software control via output bits 6 and 7 of Port C of U18. A high to either of these select bits (6 or 7) enables



+5V
1K
27K
.01µf
A
10K
2N3391
25
74122
Φ_2 (TTL)
FROM 8224
PIN 6
2 Mc
200Kc
20Kc
2Kc
1Kc
1
2
3
4
9310
POWER REQUIREMENTS
+5V @ 900MA
-5V @ 25MA
+12V @ 125MA
-12V @ 25MA
LO: ENABLE
ENABLE
BUFFER
COUNTER $\emptyset$ 1

1

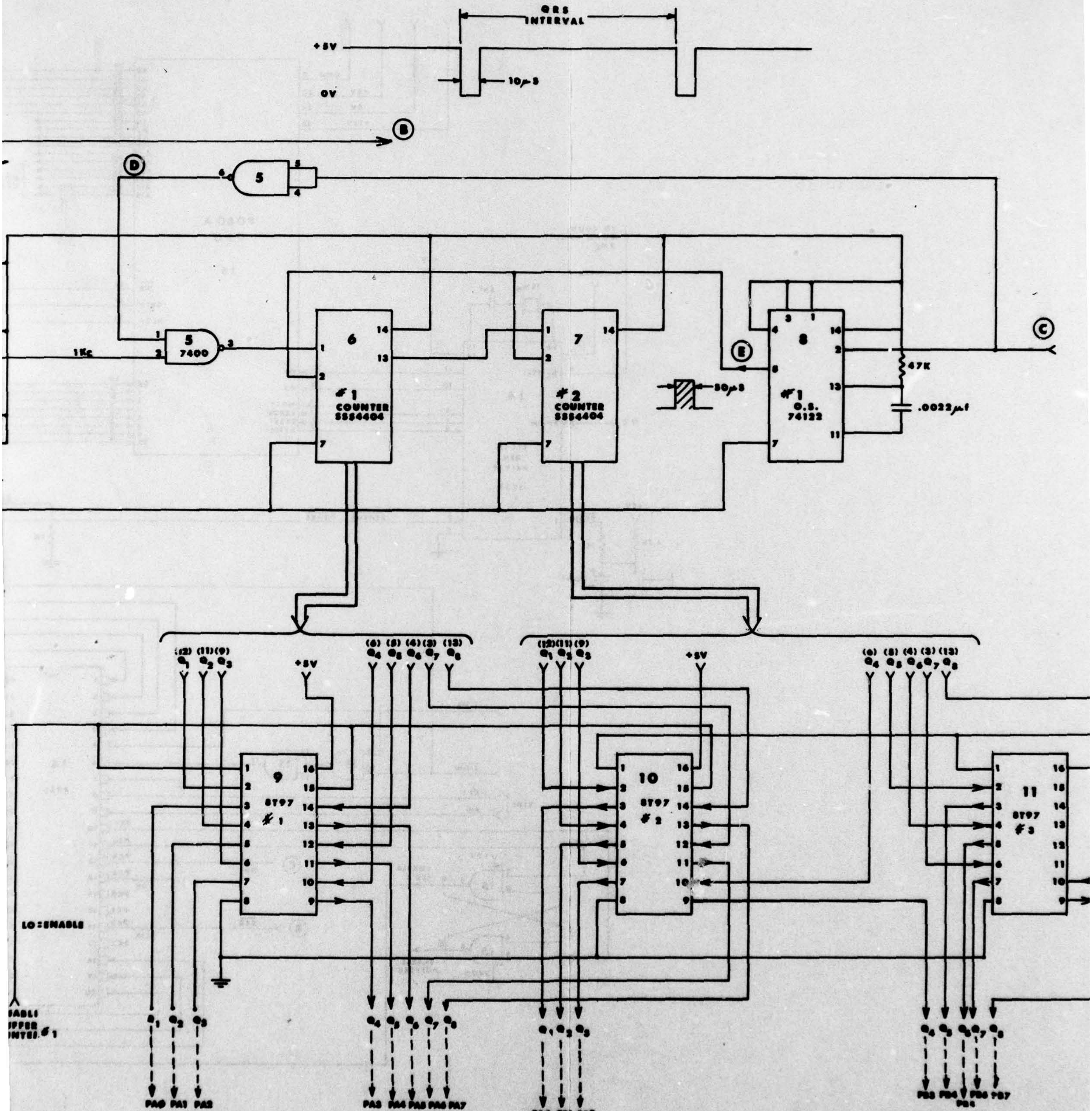


Figure 4. Timing and buffer circuitry.

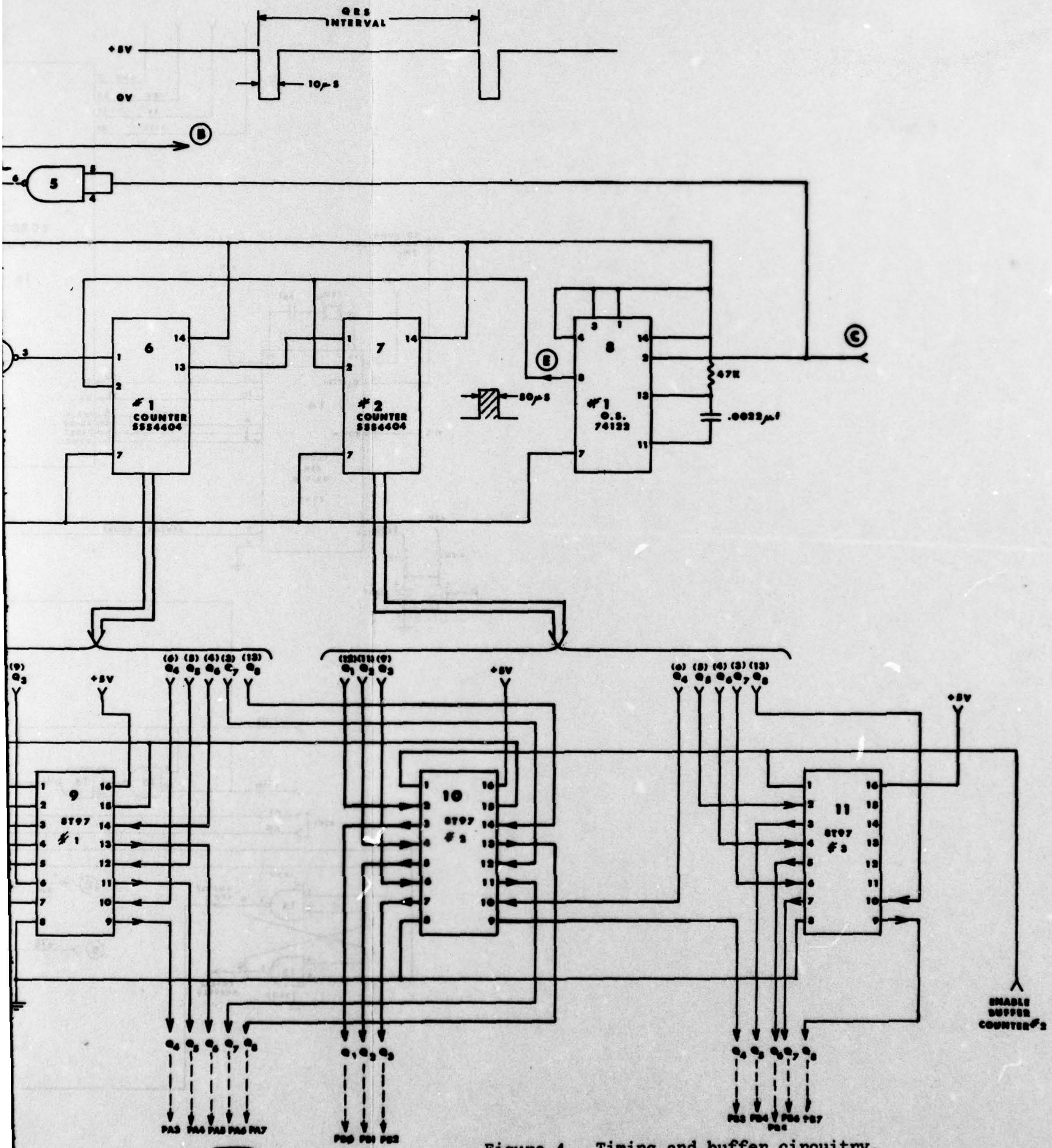


Figure 4. Timing and buffer circuitry.

2 1 3

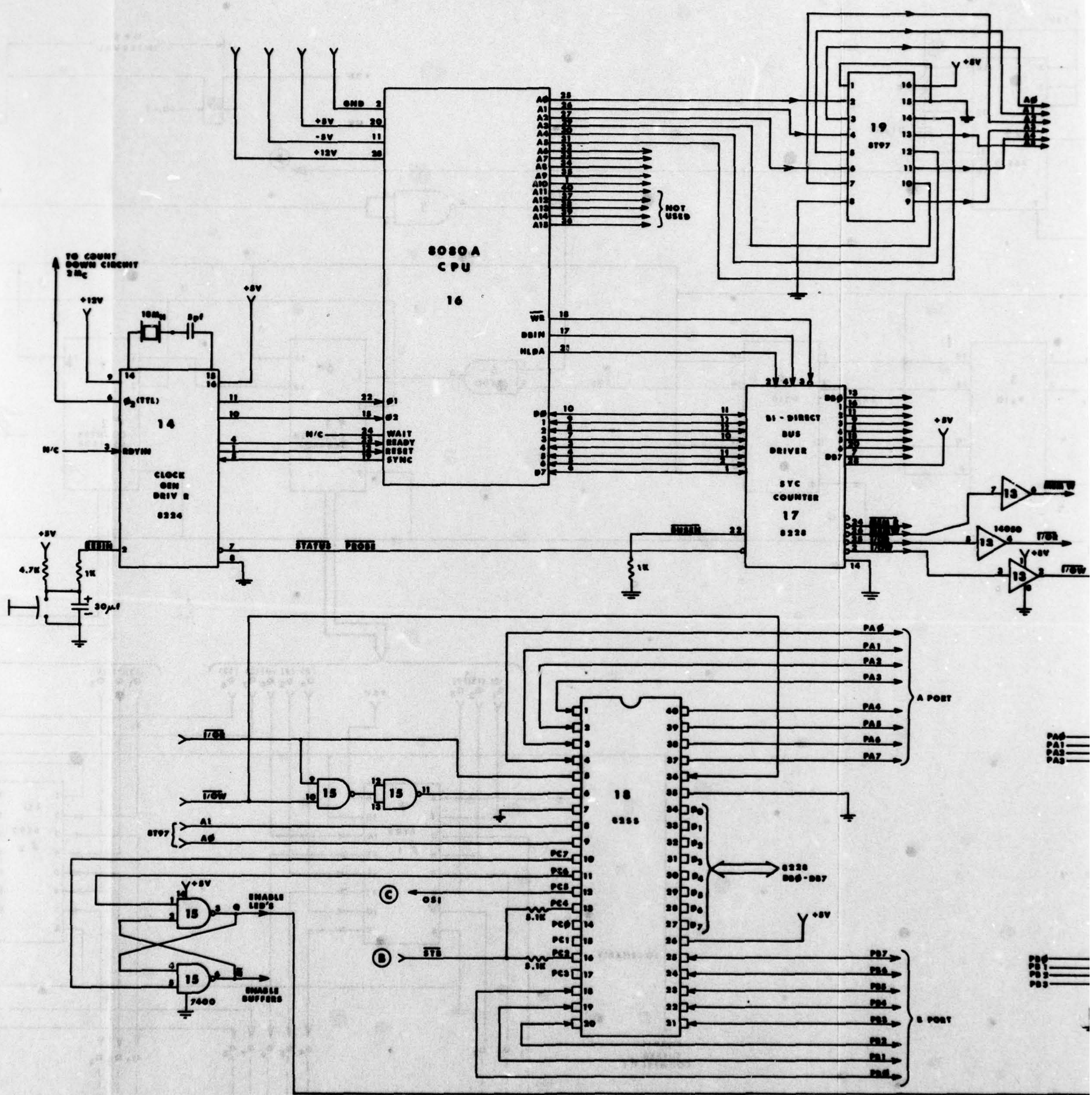


Figure 5. Micropro
microcomp

1

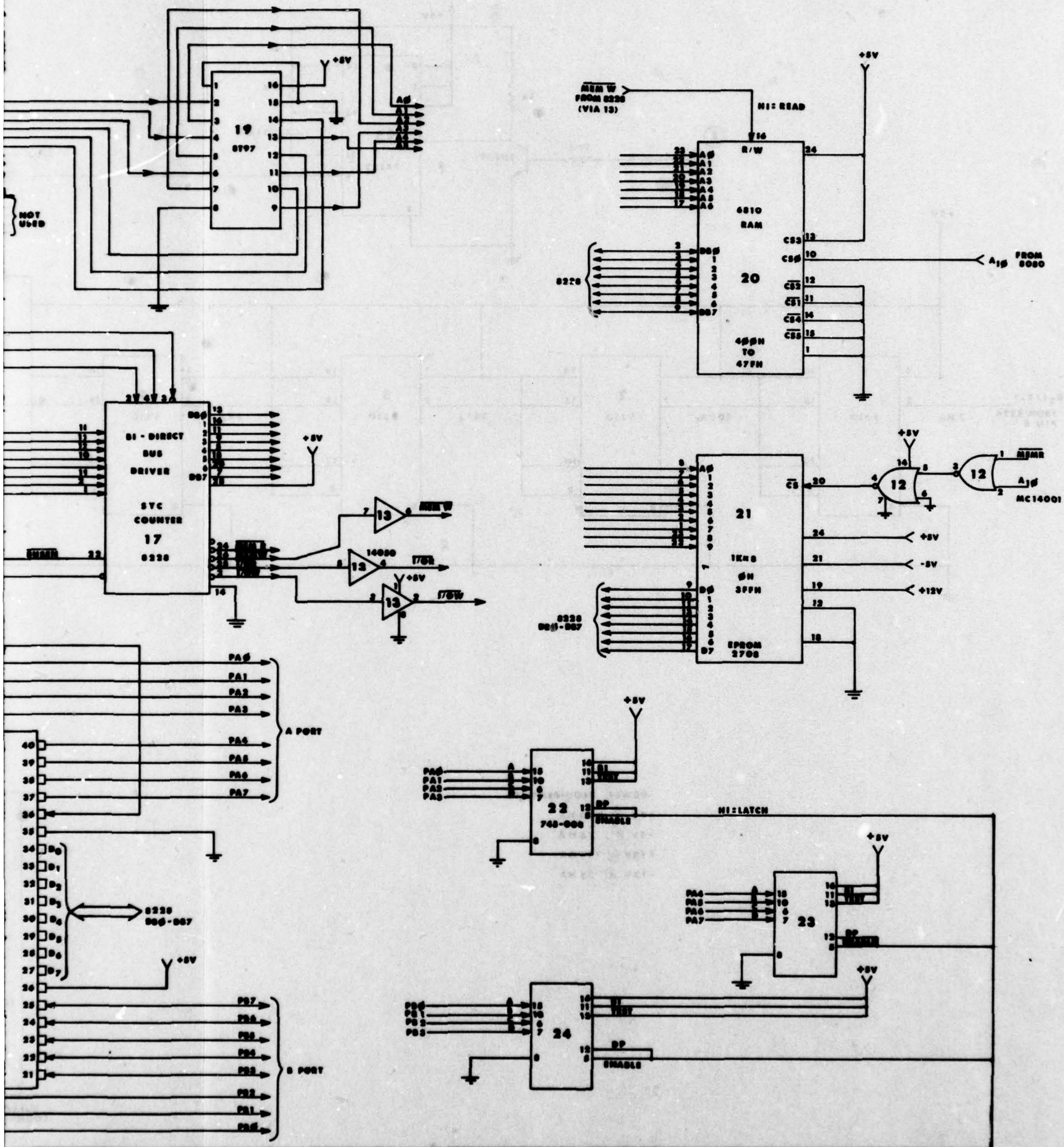


Figure 5. Microprocessor system hardware which essentially constitutes a microcomputer.

2

the corresponding device by putting a low on the appropriate enable line.

The main processing elements consist of the 8080A and a typical configuration of support chips extracted from manufacturers manuals. The 8228 uses U13 for additional control line buffering. The control program resides in the 2708 1K by 8 UV-EPROM, with a 6810 128 by 8 RAM for stack and temporary storage. Since the required memory space is so limited, elaborate 16-bit address buss decoding is unnecessary. A single bit (A-10) selects ROM at address 0 if low or RAM address 400-HEX if high. The 8255 completes the system by providing all input/output data and control connections.

Sequence Control and Timing:

The 74122 (U25) one-shot produces a negative going TTL pulse of 10 microseconds which represents the QRS signal. This QRS signal is connected to pins PC4 and PC2 of the 8255 via individual 5.1 K resistors. These resistors serve to isolate PC2 and PC4 when Port C is in output mode and have no effect in input mode. A low on this line strobes or loads counter data into input Ports A and B, respectively. With the 8255 configured in Input Mode-1, the strobe sets the input buffer full flip-flop (IBF) within the 8255. (See timing diagram in Figure 6.) The output of IBF appears at PC5 where it can be monitored by the software and also used to serve as an acknowledge signal to the counter system as follows. The normally low PC5 is inverted and applied to the other input of the gate (U5) which controls the 1 KHz clock to the counters. When PC5 goes high, input to U6 is held low, thereby stopping the count. It is essential that the count remain stable until the data can be read by the software; first Port B, then Port A. Reading of the data on Port A causes IBF to be reset (this occurs within the 8255) which allows the clock pulses to U6 to resume. However, the falling edge of IBF also triggers a one-shot (U8) which delivers a 50 microsecond pulse to reset the counters to zero. Counting thus restarts after the fall of the reset signal. The units power supply is given in Figure 7.

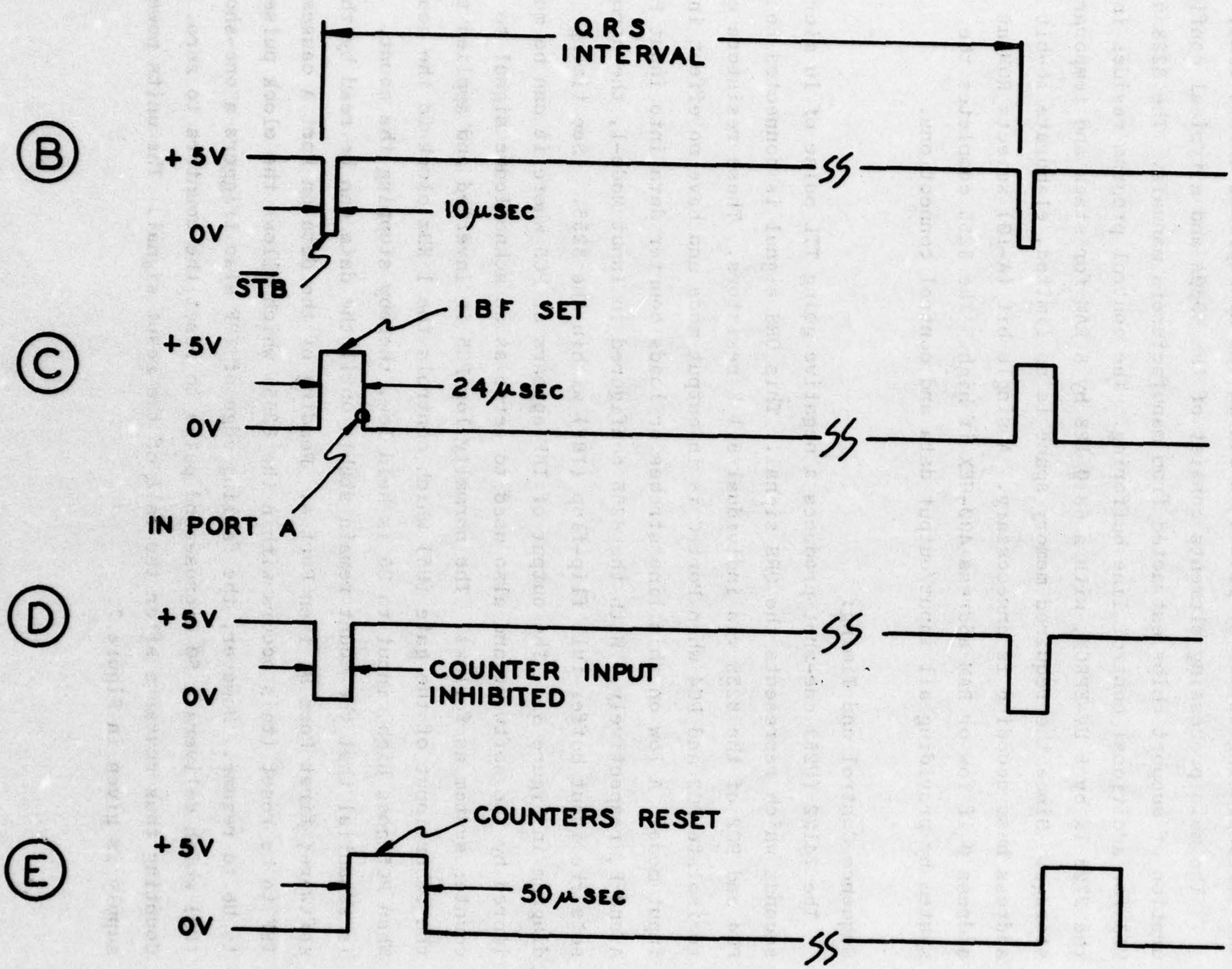
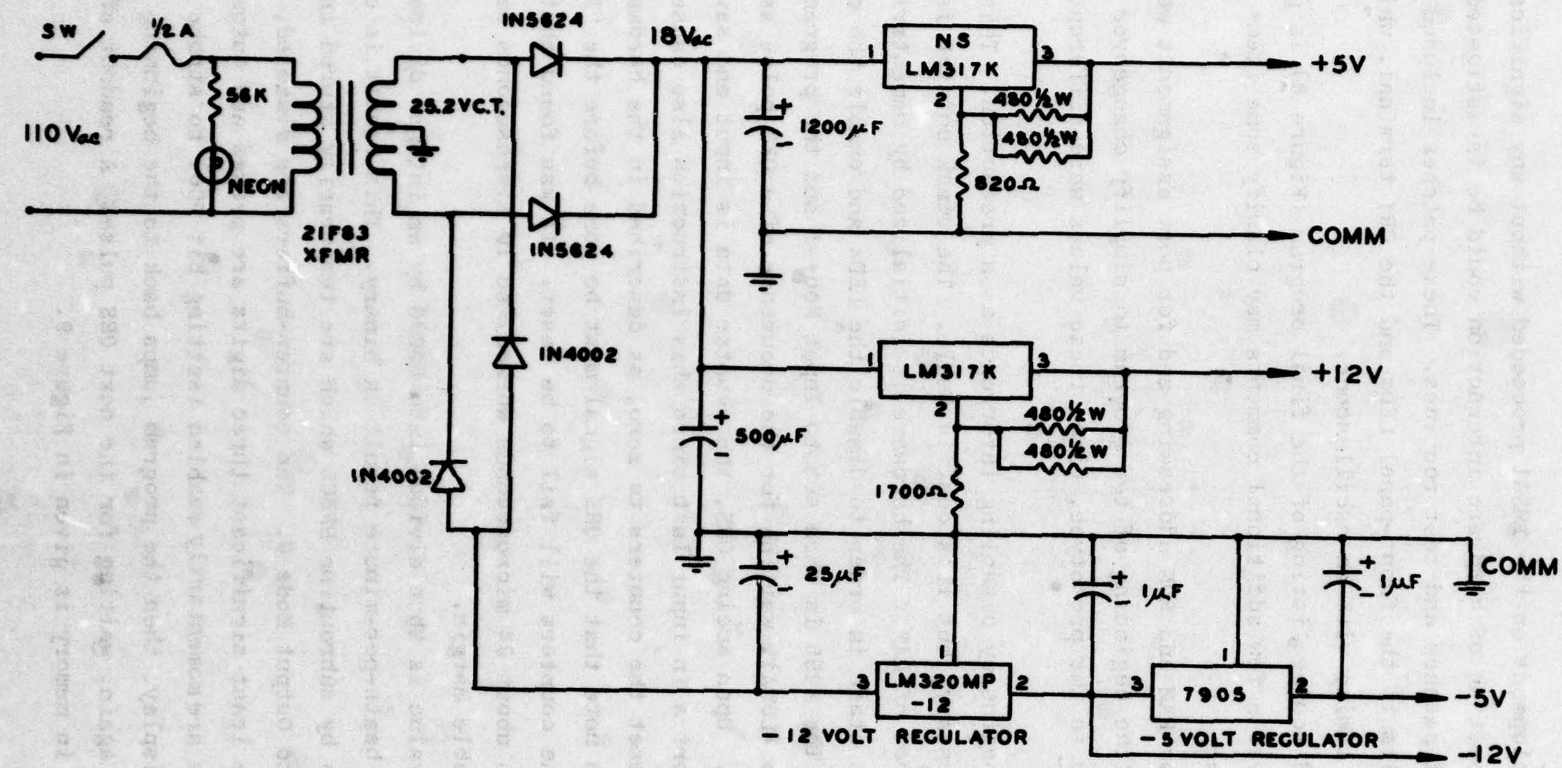


Figure 6. Timing diagram for reading and resetting the QRS interval counters.



NOTE: ALL REGULATORS ARE MOUNTED ON A SINK HEAT

Figure 7. Cardiotachometer power supply.

Software

Software development on the IMSAI proceeded without any significant difficulty as each separate section of hardware interaction could be investigated in isolation by small patches and test routines. These patches included sending intermediate results to the front-panel LEDs and the CRT terminal, while checking signal levels and timing with an oscilloscope.

The annotated source listing of the final program (Figure 8) is generally self explanatory, however a few additional comments may clarify some items of particular interest:

The values for RAM and ROM addressing and for port assignments were assigned parametrically at the beginning of the program to simplify changeover from the development system to the prototype, since these values were different for the two situations.

The program begins by disabling interrupts as a precaution. This instruction should not be necessary, but it's cost is small. The stack pointer is then set to an available area of RAM. The hardware is initialized by momentarily setting the 8255 to output state in order to disable the LEDs and enable the counter-buffer circuits. The 8255 is then set to Input Mode-1 and the program enters a test-and-wait loop (LOPA), waiting for the occurence of a QRS pulse as detected by PC5 going high. Upon seeing QRS, the counter data is input and saved, first Port B then A. Port A in input last since this instruction also causes a chain of events which reset the counters to zero, as described in the hardware section. It is important to note that the QRS signal must be gone before the "In Port A" instruction, or the counters will fail to be reset. It was found that any QRS pulse length up to about 24 microseconds worked, so 1Ø microseconds was chosen to provide a reliable margin.

The counter value is then divided into 6ØØØØ by an integer divide subroutine (DIV), giving the beats-per-minute result in binary. This result is converted to five BCD digits by subroutine BNBCD which are temporarily stored in RAM. The 8255 is then set to Output Mode Ø. The counter-buffers are disabled, and the BCD values for the least significant three digits are packed and output to Ports A and B. The LEDs are momentarily enabled (setting bit PC6) to strobe the new values into the display, then the program jumps back to the beginning of the loop to start all over again, waiting for the next QRS pulse. A readout of the EPROM's assembled program in memory is given in Figure 9.

```
0010 * CARDS
0020 * CARDIOTACHOMETER.
0030 * HEARTRATE DISPLAYED IN DECIMAL BEATS PER MINUTE.
0040 *
0050 *
0060 * MEMORY ASSIGNMENTS
0070 ROM     EQU     0
0080 RAM     EQU     400H
0090 *
0100 * 8255 PORT ASSIGNMENTS:
0110 PORTA   EQU     0
0120 PORTB   EQU     PORTA+1
0130 PORTC   EQU     PORTA+2
0140 CNTRL   EQU     PORTA+3   ; CONTROL PORT
0150 *
0160 PSW     EQU     6
0170 SP      EQU     6
0180 *
0190         ORG     ROM
0200 *
0210 LOOP:   DI              ; DISABLE INTERRUPTS
0220         LXI     SP,STACK  ; RESET STACK POINTER
0230 *
0240         MVI     A,80H   ; SET 8255 MODE 0, ALL OUTPUT
0250         OUT     CNTRL
0260         MVI     A,80H
0270         OUT     PORTC   ; ENABLE BUFFERS (SET BIT 7)
0280 *
0290         MVI     A,0BFH  ; SET 8255 MODE 1, ALL INPUT
0300         OUT     CNTRL
0310 *
0320 * EVENT PULSE SHOULD BE NO LONGER THAN APPROX.
0330 * 20 USEC. MUST BE COMPLETED BEFORE THE
0340 * INSTRUCTION - < IN PORTA > , BELOW.
0350 LOPA:   IN      PORTC
0360         ANI     20H     ; BIT 5
0370         JZ      LOPA    ; WAIT FOR EVENT PULSE
0380 *
0390 * READ MILLISECOND COUNT SINCE LAST EVENT
0400 * AND SAVE IN DE FOR COMPUTING EVENTS PER MINUTE.
0410 *
0420         IN      PORTB   ; MSB'S
0430         MOV     D,A
0440         IN      PORTA   ; LSB'S
0450         MOV     E,A
0460 *E
```

Figure 8. Annotated source listing of cardiotachometer program

```
* COMPUTE BEATS PER MINUTE..
* NUMBER OF MILLISECONDS PER MINUTE (60,000) DIVIDED BY
* NUMBER OF MILLISECONDS PER EVENT INTERVAL
* ( I.E. NO. OF MSEC. COUNTS PER INTERVAL)
* GIVES NUMBER OF EVENTS PER MINUTE.
*
        LXI     B,60000
        CALL    DIV     ; OBTAIN QUOTIENT IN BC
        MOV     D,B     ; MOVE TO DE
        MOV     E,C
*
        LXI     H,NUM1   ; SET NUMERAL BUFFER
        CALL    BNBCD
*
        MVI     A,80H   ; SET MODE 0, ALL OUTPUT
        OUT     CNTRL
        MVI     A,0
        OUT     PORTC   ; DISABLE COUNTER BUFFERS.
*
        LDA     NUM3
        ANI     0FH
        MOV     B,A
        LDA     NUM4
        RLC
        RLC
        RLC
        RLC
        ANI     0F0H
        ORA     B       ; PACK FIRST 2 DIGITS
        OUT     PORTA
        LDA     NUM5
        OUT     PORTB
*
        MVI     A,40H   ; ENABLE LED'S
        OUT     PORTC
        MVI     A,0     ; DISABLE LED'S
        OUT     PORTC
*
        JMP     LOOP
*E
```

Figure 8. Continued

```
0870 * 16-BIT DIVIDE ROUTINE
0880 * ENTER WITH DIVIDEND IN BC, DIVISOR IN DE
0890 * EXIT WITH QUOTIENT IN BC
0900 * IGNORE REMAINDER.
0910 *
0920 DIV     MOV    A,D
0930         CMA
0940         MOV    D,A
0950         MOV    A,E
0960         CMA
0970         MOV    E,A
0980         INX    D
0990         LXI    H,0
1000         MVI    A,17
1010 DV0     PUSH   H
1020         DAD    D
1030         JNC    DV1
1040         XTHL
1050 DV1     POP    H
1060         PUSH   PSW
1070         MOV    A,C
1080         RAL
1090         MOV    C,A
1100         MOV    A,B
1110         RAL
1120         MOV    B,A
1130         MOV    A,L
1140         RAL
1150         MOV    L,A
1160         MOV    A,H
1170         RAL
1180         MOV    H,A
1190         POP    PSW
1200         DCR    A
1210         JNZ    DV0
1220         RET
1230 *E
```

Figure 8. Continued

```
* CONVERT BINARY TO BCD
* ENTER WITH ADDRESS OF STORAGE AREA IN HL
* AND BINARY VALUE IN DE
*
BNBCD   PUSH    PSW
        PUSH    B
        PUSH    D
        PUSH    H
        XCHG
        LXI     B,-10000
        CALL    DECNO
        LXI     B,-1000
        CALL    DECNO
        LXI     B,-100
        CALL    DECNO
        LXI     B,-10
        CALL    DECNO
        MOV     A,L
        STAX    D
        POP     H
        POP     D
        POP     B
        POP     PSW
        RET
*
DECNO   XRA     A
        PUSH    D
        MOV     E,L
        MOV     D,H
        INR     A
        DAD     B
        JC      DECNO+2
        DCR     A
        MOV     L,E
        MOV     H,D
        POP     D
        STAX    D
        INX     D
        RET
*
TIP     EQU     $       ; END OF ROM AREA
        COM     TIP
*E
```

Figure 8. Continued

```
1670 *    RAM AREA:
1680         ORG     RAM
1690 *
1700 * DIGIT STORAGE
1710 NUM1:   DB      0
1720 NUM2:   DB      0
1730 NUM3:   DB      0
1740 NUM4:   DB      0
1750 NUM5:   DB      0
1760 *
1770 * ASSIGN STACK SPACE
1780         DS      20H
1790 STACK:  DW      0
1800 *E
```

Figure 8. Continued

```
0000: F3 31 25 04 3E 80 D3 03 3E 80 D3 02 3E BF D3 03
0010: DB 02 E6 20 CA 10 00 DB 01 57 DB 00 5F 01 60 EA
0020: CD 55 00 50 59 21 00 04 CD 7B 00 3E 80 D3 03 3E
0030: 00 D3 02 3A 02 04 E6 0F 47 3A 03 04 07 07 07 07
0040: E6 F0 B0 D3 00 3A 04 04 D3 01 3E 40 D3 02 3E 00
0050: D3 02 C3 00 00 7A 2F 57 7B 2F 5F 13 21 00 00 3E
0060: 11 E5 19 D2 67 00 E3 E1 F5 79 17 4F 78 17 47 7D
0070: 17 6F 7C 17 67 F1 3D C2 61 00 C9 F5 C5 D5 E5 EB
0080: 01 F0 D8 CD 9F 00 01 18 FC CD 9F 00 01 9C FF CD
0090: 9F 00 01 F6 FF CD 9F 00 7D 12 E1 D1 C1 F1 C9 AF
00A0: D5 5D 54 3C 09 DA A1 00 3D 6B 62 D1 12 13 C9
```

Figure 9. Hexadecimal readout of the assembled program

CONCLUSIONS

The microprocessor based cardiotachometer described here has proved to be very accurate (due to its crystal controlled time base) and extremely reliable, with no failures or tendency to instability after extensive bench testing. It requires no calibration, and in packaged form would be expected to operate without problems indefinitely. Changes and improvements, such as adding a digital to analog converter to provide a signal for driving a chart recorder, become immediately obvious. Using a microprocessor as the basis for any instrument makes it possible to add large complexity in behavior with relatively small changes in hardware.

As the project progressed, the nature and operation of each of the circuit elements became increasingly familiar. The 8255 programmable peripheral interface was particularly confusing however, and the Intel 8255A Application Note AP-15 in addition to the Intel Systems Users Manual was required. The experience gained with this instrument has provided a great confidence in the use of microprocessor elements as integral parts of any instrumentation of control system.

APPENDIX A

Cartest 2 Program

The following listing is a test program called CARTEST2 implemented on a UV PROM used to check out the microprocessor system hardware for improper operation. First, the number 124 is displayed. Then all digits are zeroed. Next, each digit in turn, displays the count from zero to nine.

```
* 7JUL78 1300
* CARTEST2.A
* CARDIOTACHOMETER TEST ROUTINE
*
* MEMORY ASSIGNMENTS
ROM     EQU     0
RAM     EQU     400H
*
* PORT ASSIGNMENTS
PORTA   EQU     0
PORTB   EQU     PORTA+1
PORTC   EQU     PORTA+2
CNTRL   EQU     PORTA+3   ; CONTROL PORT
*
PSW     EQU     6
SP      EQU     6
*
        ORG     ROM
*
LOOP:   DI              ; DISABLE INTERRUPTS
        LXI     SP,STACK   ; RESET STACK POINTER
*
        MVI     A,80H   ; SET MODE 0, ALL OUTPUT
        OUT     CNTRL
*
        MVI     A,21H
        OUT     PORTA
        MVI     A,4
        OUT     PORTB
*
        CALL    STROB
        CALL    DLY
        CALL    DLY
        CALL    DLY
*
        MVI     A,0
        OUT     PORTA
        OUT     PORTB
        CALL    DLY
*
        MVI     B,10
        MVI     C,0
LOP1:   CALL    DLY
        MOV     A,C
        OUT     PORTA
        CALL    STROB
```

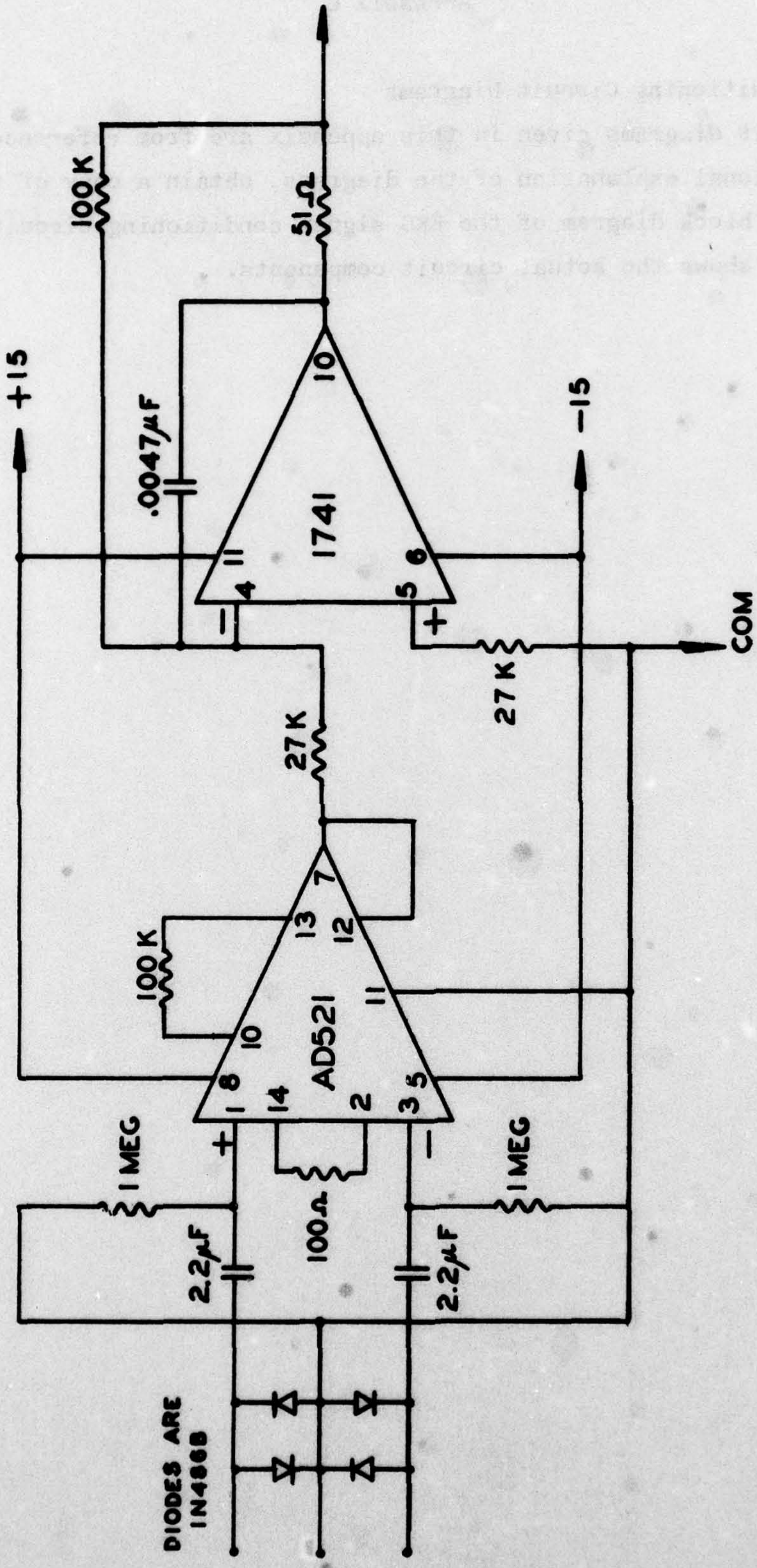
```
0470            INR     C
0480            DCR     B
0490            JNZ     LOP1
0500 *E
0510            MVI     B,10
0520            MVI     C,0
0530 LOP2:      CALL    DLY
0540            MOV     A,C
0550            RLC
0560            RLC
0570            RLC
0580            RLC
0590            OUT     PORTA
0600            CALL    STROB
0610            INR     C
0620            DCR     B
0630            JNZ     LOP2
0640 *
0650            MVI     B,10
0660            MVI     C,0
0670 LOP3:      CALL    DLY
0680            MOV     A,C
0690            OUT     PORTB
0700            CALL    STROB
0710            INR     C
0720            DCR     B
0730            JNZ     LOP3
0740 *
0750            JMP     LOOP
0760 *
0770 *
0780 STROB:     MVI     A,40H   ; ENABLE LED'S
0790            OUT     PORTC
0800            MVI     A,80H
0810            OUT     PORTC   ; DISABLE LED'S, ENABLE COUNT BUFFERS.
0820            RET
0830 *
0840 * DELAY 1 SECOND
0850 DLY:       LXI     H,1000
0860            PUSH    PSW
0870 DLY0:      DCX     H
0880            MVI     A,132
0890 DLY1:      DCR     A
0900            JNZ     DLY1
0910            MOV     A,H
0920            ORA     L
0930            JNZ     DLY0
0940            POP     PSW
0950            RET
0960 *
0970 TIP        EQU     $
0980            COM     TIP
0990 *
1000 * RAM AREA
1010            ORG     RAM
```

```
            DS      20H
     STACK: DW      0
     *E
     *
```

APPENDIX B

Electrocardiogram Amplifier

An electrocardiogram amplifier that can be used to obtain an EKG input for the cardiotachometer is given below. Circuitry is the courtesy of Mr. Donald McCollor, Raytheon Service Company.

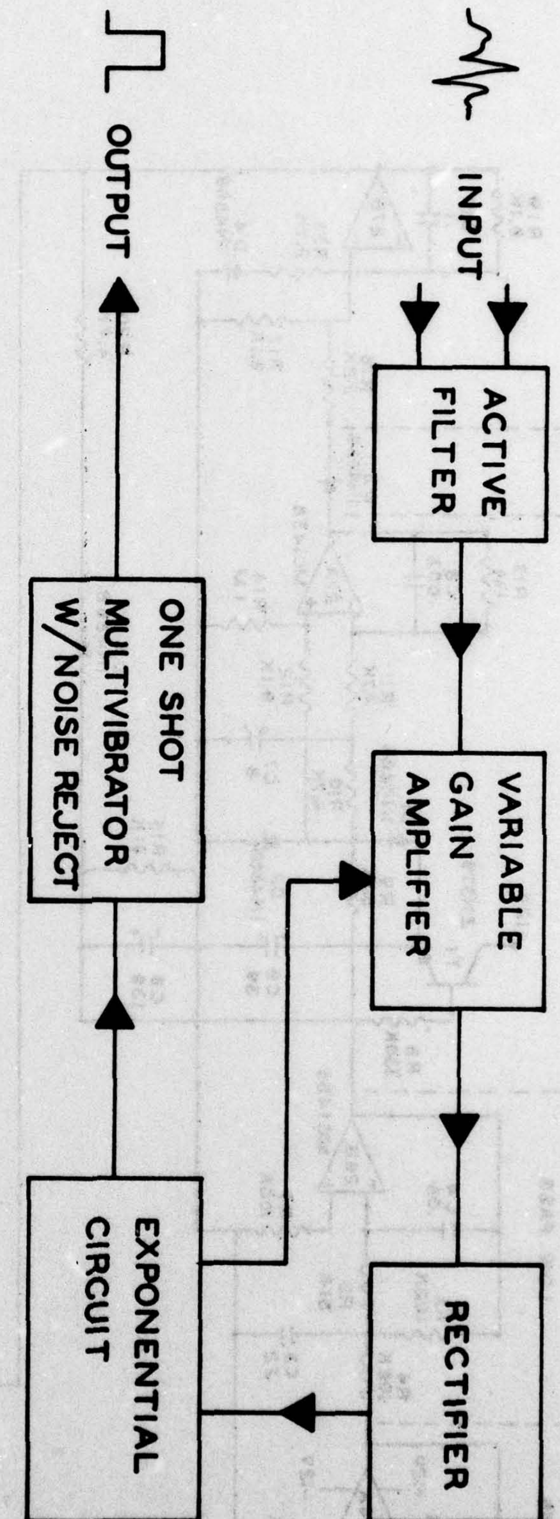


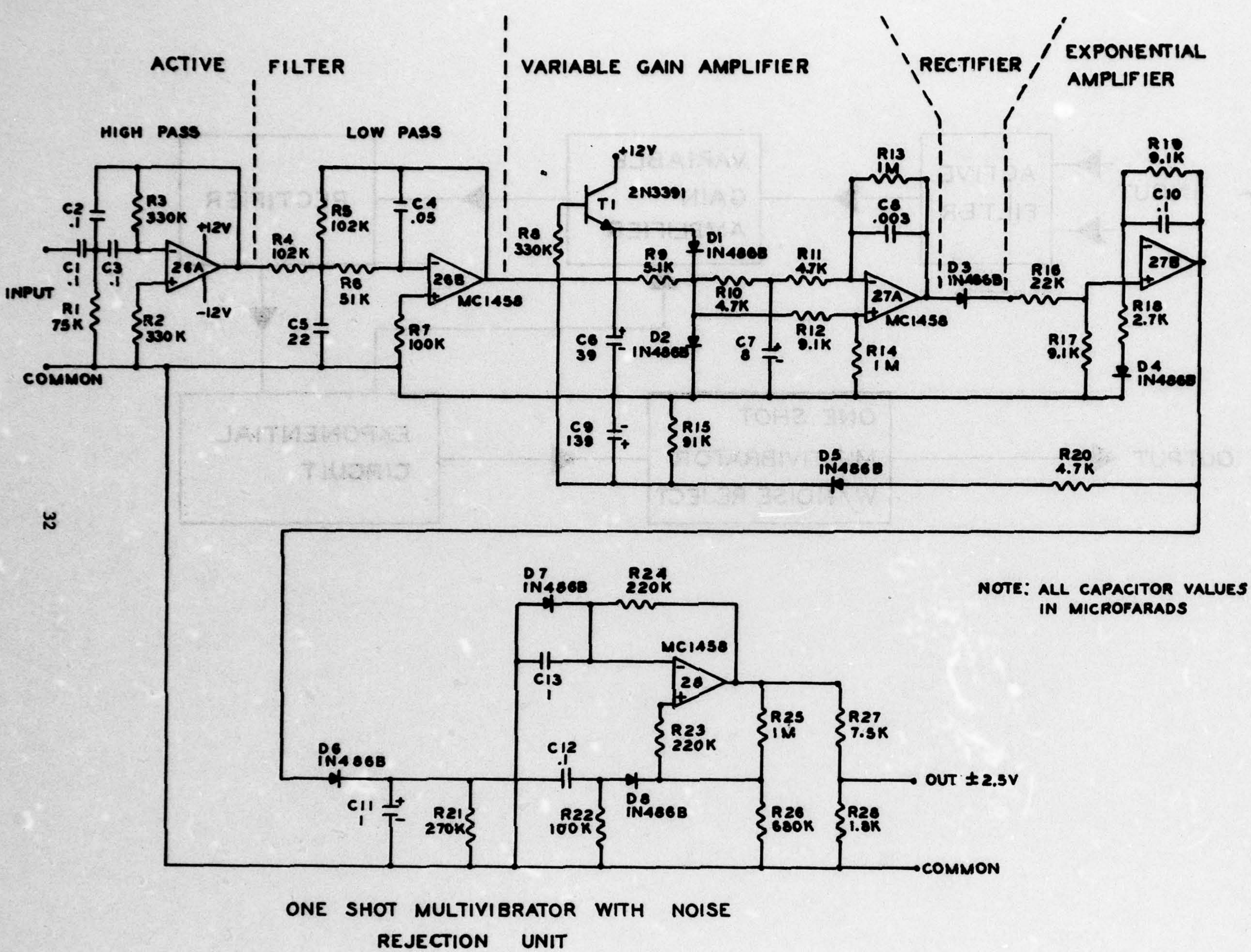
+15
-15
COM
100 K
51 Ω
.0047μF
1741
27 K
27 K
AD521
100 K
1 MEG
1 MEG
2.2μF
2.2μF
100Ω
DIODES ARE

APPENDIX C

EKG Signal Conditioning Circuit Diagrams

The circuit diagrams given in this appendix are from reference 1. For a detailed functional explanation of the diagrams, obtain a copy of the above report. The first is a block diagram of the EKG signal conditioning circuitry. The second diagram shows the actual circuit components.


INPUT
ACTIVE FILTER
VARIABLE GAIN AMPLIFIER
RECTIFIER
EXPONENTIAL CIRCUIT
ONE SHOT MULTIVIBRATOR W/NOISE REJECT
OUTPUT


ACTIVE FILTER
HIGH PASS
LOW PASS
VARIABLE GAIN AMPLIFIER
RECTIFIER
EXPONENTIAL AMPLIFIER
INPUT
COMMON
C2 .1
C1 .1
C3 .1
R1 75K
R3 330K
R2 330K
26A
+12V
-12V
R4 102K
R5 102K
R6 51K
C4 .05
C5 22
26B
MC1458
R7 100K
R8 330K
+12V
T1
2N3391
D1 IN486B
D2 IN486B
R9 5.1K
R10 4.7K
R11 4.7K
R12 9.1K
C6 39
C7 8
C9 139
R15 91K
D5 IN486B
R13 1M
C8 .003
27A
MC1458
R14 1M
D3 IN486B
R16 22K
R17 9.1K
R18 2.7K
D4 IN486B
R19 9.1K
C10 .1
27B
R20 4.7K
NOTE: ALL CAPACITOR VALUES IN MICROFARADS
D7 IN486B
R24 220K
MC1458
28
C13 1
R23 220K
R25 1M
R27 7.5K
C12 .1
D6 IN486B
D8 IN486B
C11 1
R21 270K
R22 100K
R26 680K
R28 1.8K
OUT ±2.5V
COMMON
ONE SHOT MULTIVIBRATOR WITH NOISE REJECTION UNIT

REFERENCES

Marko, Adolf R., et al., An Improved Cardiotachometer Input Circuit for Heart Rate Determination, AMRL-TR-73-50, Wright Patterson Air Force Base, Ohio, September 1973.

*U.S.Government Printing Office: 1979 — 657-002/697